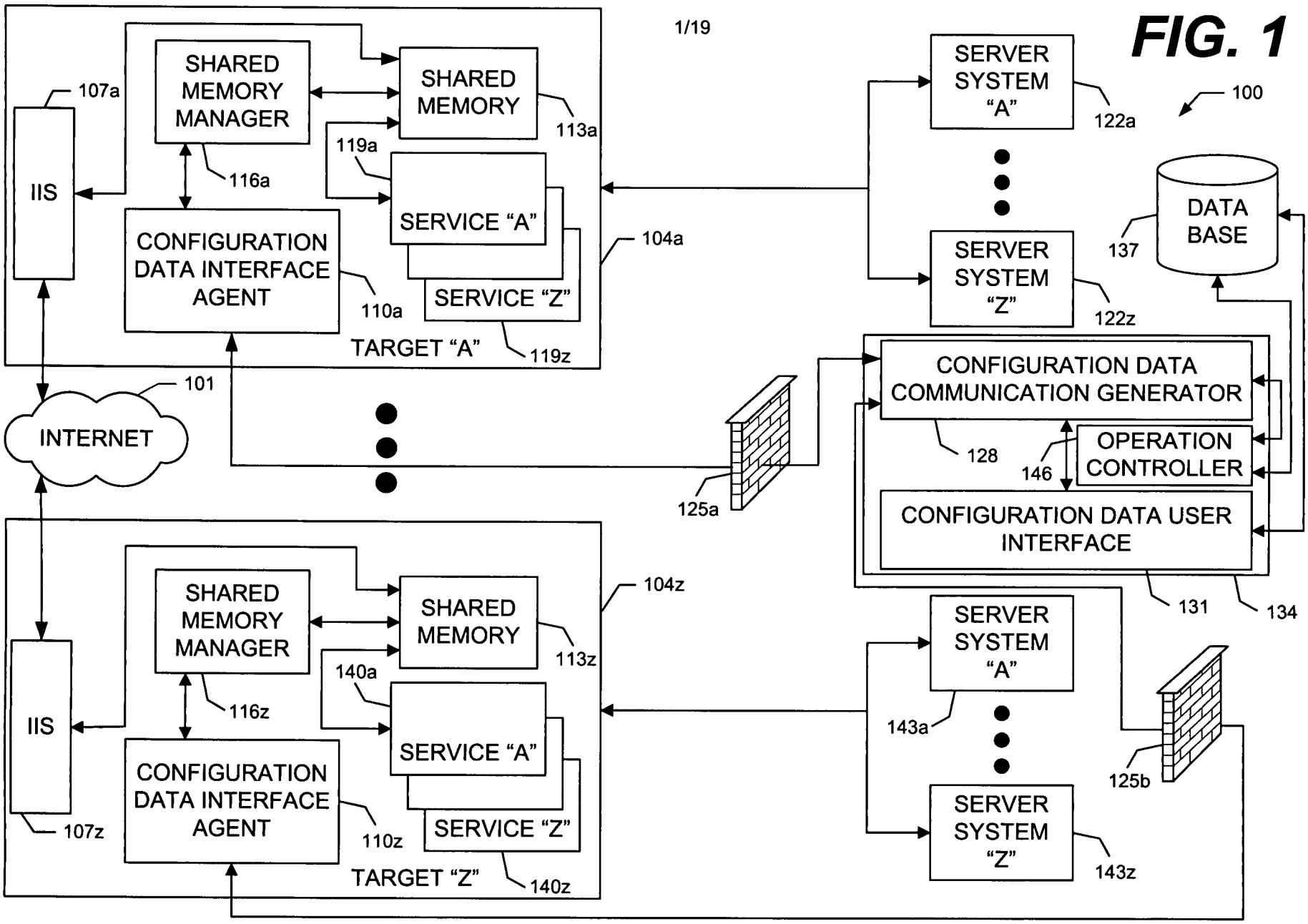
FIG. 1


SHARED MEMORY MANAGER
SHARED MEMORY
113a
107a
IIS
116a
119a
SERVICE "A"
CONFIGURATION DATA INTERFACE AGENT
110a
SERVICE "Z"
119z
TARGET "A"
104a
INTERNET
101
SERVER SYSTEM "A"
122a
100
DATA BASE
137
SERVER SYSTEM "Z"
122z
CONFIGURATION DATA COMMUNICATION GENERATOR
128
OPERATION CONTROLLER
146
CONFIGURATION DATA USER INTERFACE
131
134
125a
SHARED MEMORY MANAGER
SHARED MEMORY
113z
104z
140a
IIS
116z
SERVICE "A"
CONFIGURATION DATA INTERFACE AGENT
110z
SERVICE "Z"
140z
TARGET "Z"
107z
SERVER SYSTEM "A"
143a
SERVER SYSTEM "Z"
143z
125b

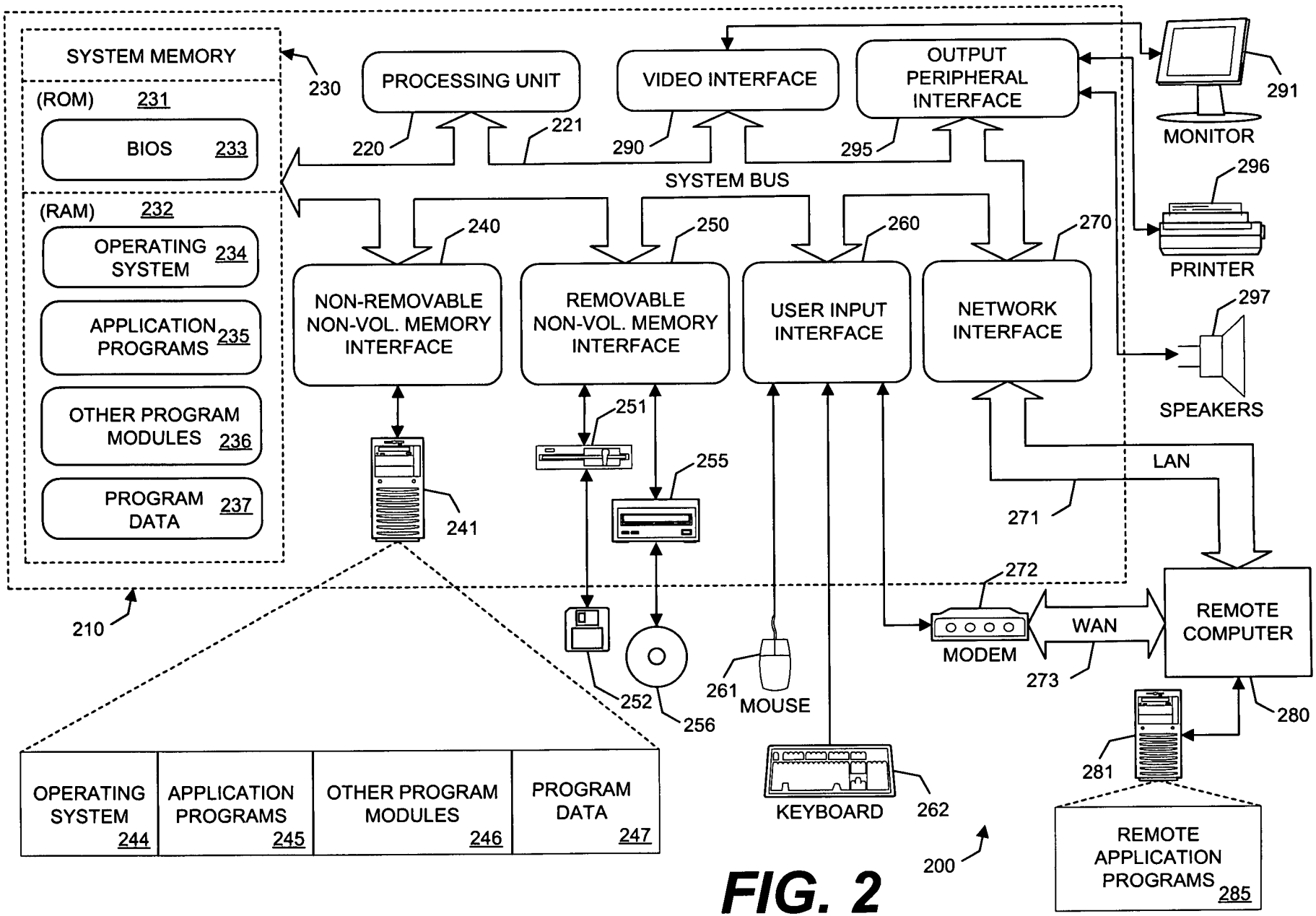

SYSTEM MEMORY
230
(ROM) 231
BIOS 233
(RAM) 232
OPERATING SYSTEM 234
APPLICATION PROGRAMS 235
OTHER PROGRAM MODULES 236
PROGRAM DATA 237
PROCESSING UNIT
220
221
VIDEO INTERFACE
290
OUTPUT PERIPHERAL INTERFACE
295
SYSTEM BUS
240
250
260
270
NON-REMOVABLE NON-VOL. MEMORY INTERFACE
REMOVABLE NON-VOL. MEMORY INTERFACE
USER INPUT INTERFACE
NETWORK INTERFACE
241
251
255
252
256
261
MOUSE
KEYBOARD
262
MONITOR
291
PRINTER
296
SPEAKERS
297
LAN
271
MODEM
272
WAN
273
REMOTE COMPUTER
280
281
REMOTE APPLICATION PROGRAMS 285
210
200
OPERATING SYSTEM 244
APPLICATION PROGRAMS 245
OTHER PROGRAM MODULES 246
PROGRAM DATA 247

FIG. 2

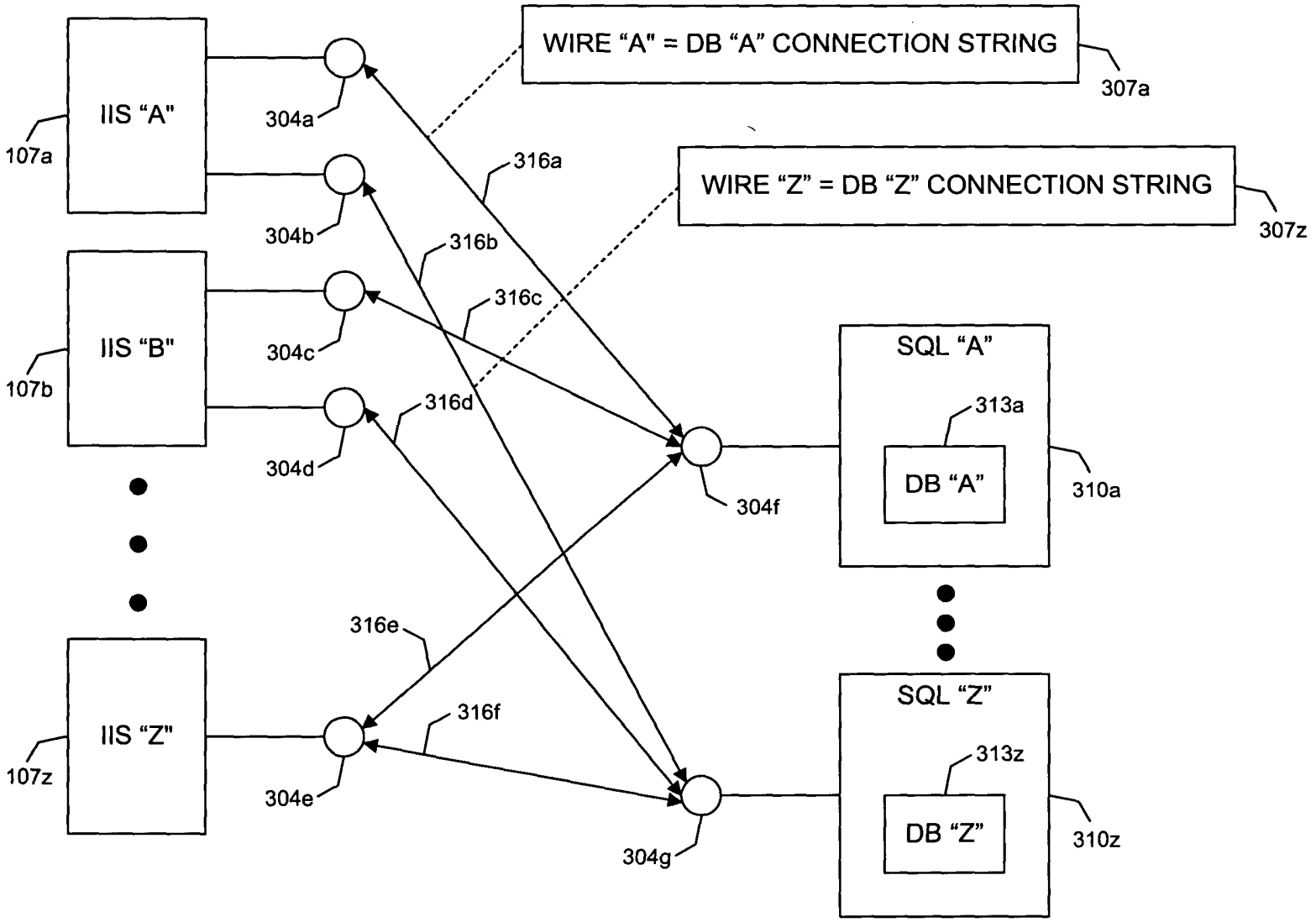

WIRE "A" = DB "A" CONNECTION STRING
307a
WIRE "Z" = DB "Z" CONNECTION STRING
307z
IIS "A"
107a
IIS "B"
107b
IIS "Z"
107z
304a
304b
304c
304d
304e
304f
304g
316a
316b
316c
316d
316e
316f
SQL "A"
313a
DB "A"
310a
SQL "Z"
313z
DB "Z"
310z

FIG. 3

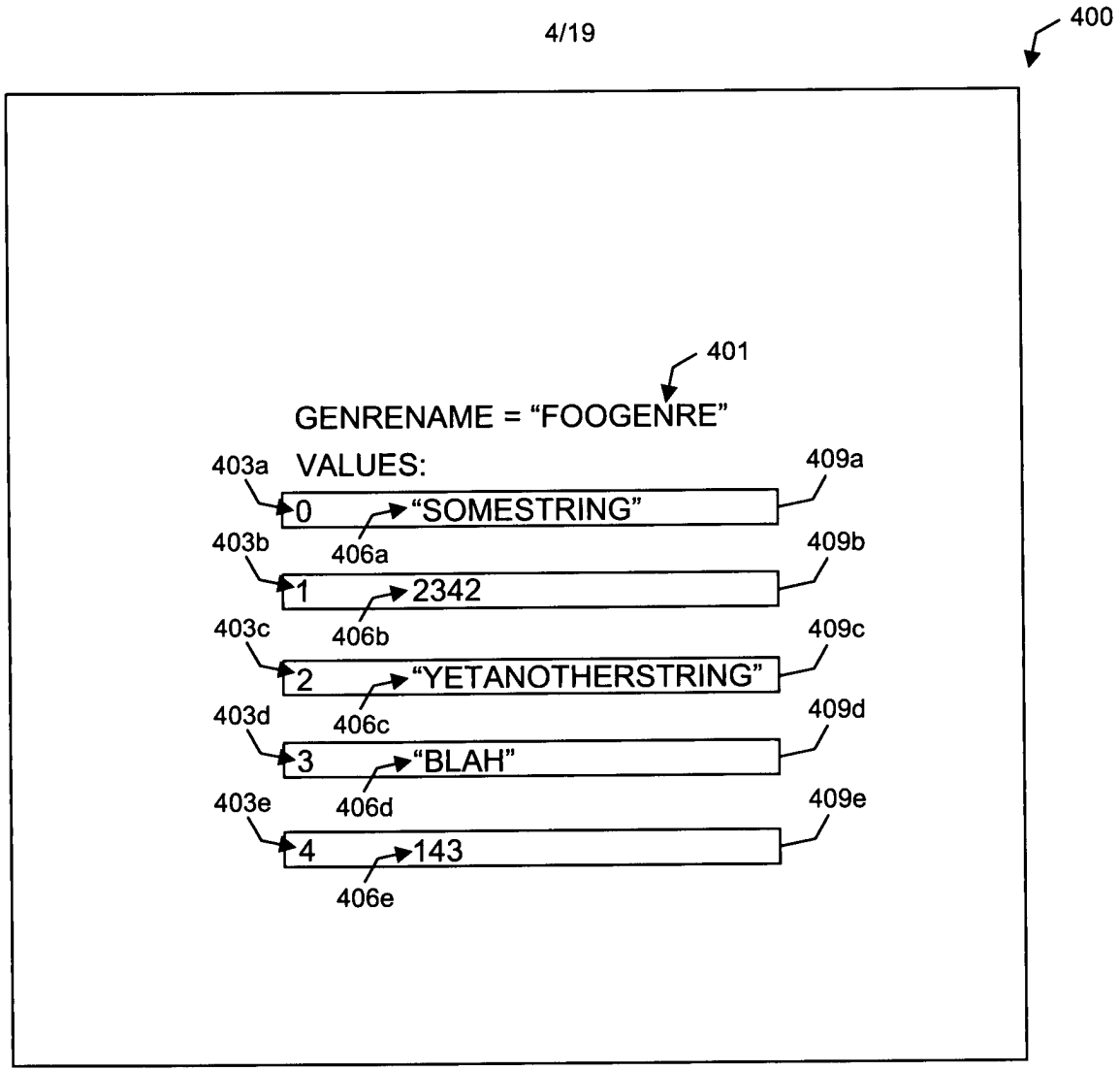
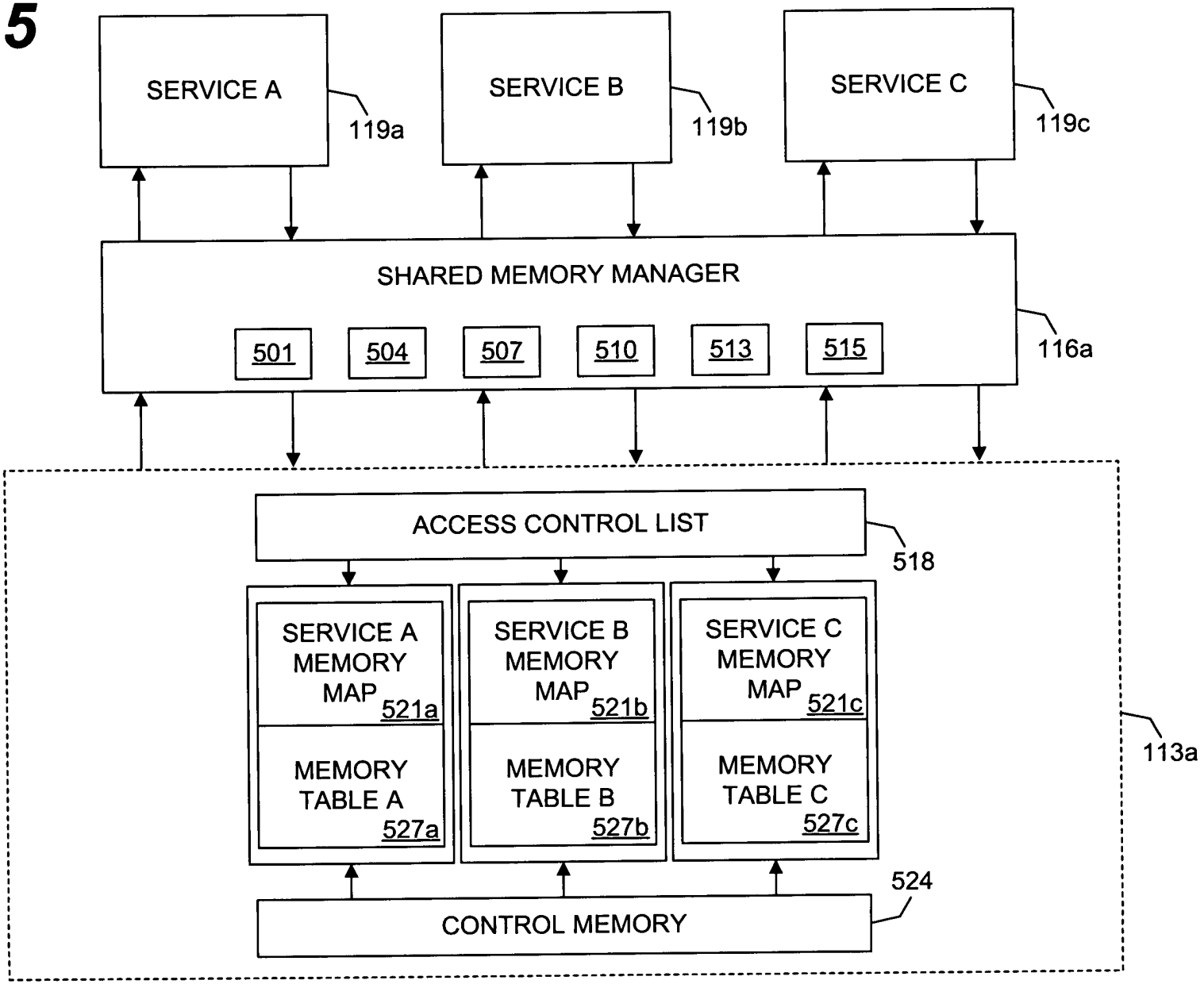

400
401
GENRENAME = “FOOGENRE”
VALUES:
403a
0
“SOMESTRING”
406a
409a
403b
1
2342
406b
409b
403c
2
“YETANOTHERSTRING”
406c
409c
403d
3
“BLAH”
406d
409d
403e
4
143
406e
409e

FIG. 4

FIG. 5


SERVICE A
119a
SERVICE B
119b
SERVICE C
119c
SHARED MEMORY MANAGER
501
504
507
510
513
515
116a
ACCESS CONTROL LIST
518
SERVICE A MEMORY MAP 521a
SERVICE B MEMORY MAP 521b
SERVICE C MEMORY MAP 521c
MEMORY TABLE A 527a
MEMORY TABLE B 527b
MEMORY TABLE C 527c
CONTROL MEMORY
524
113a

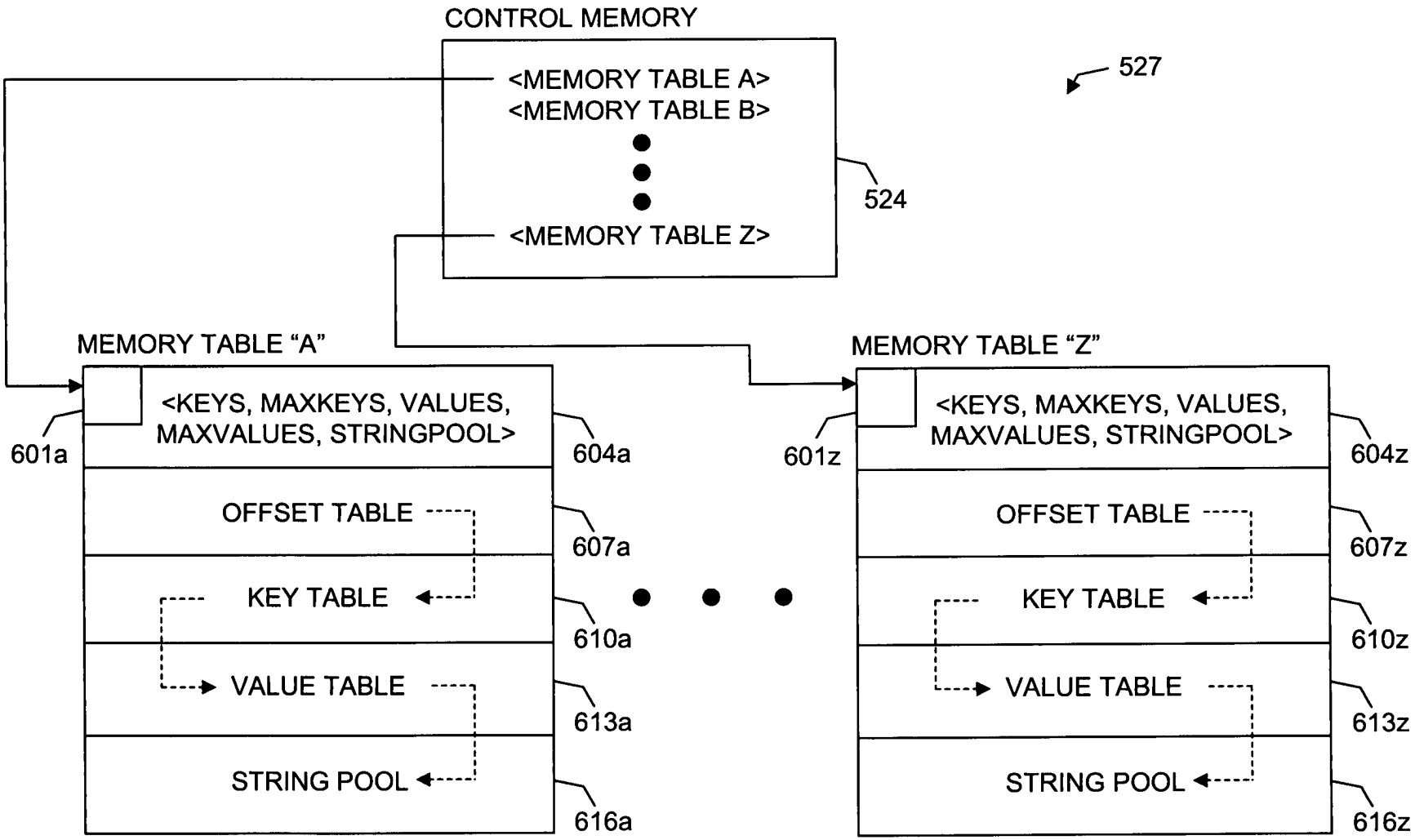
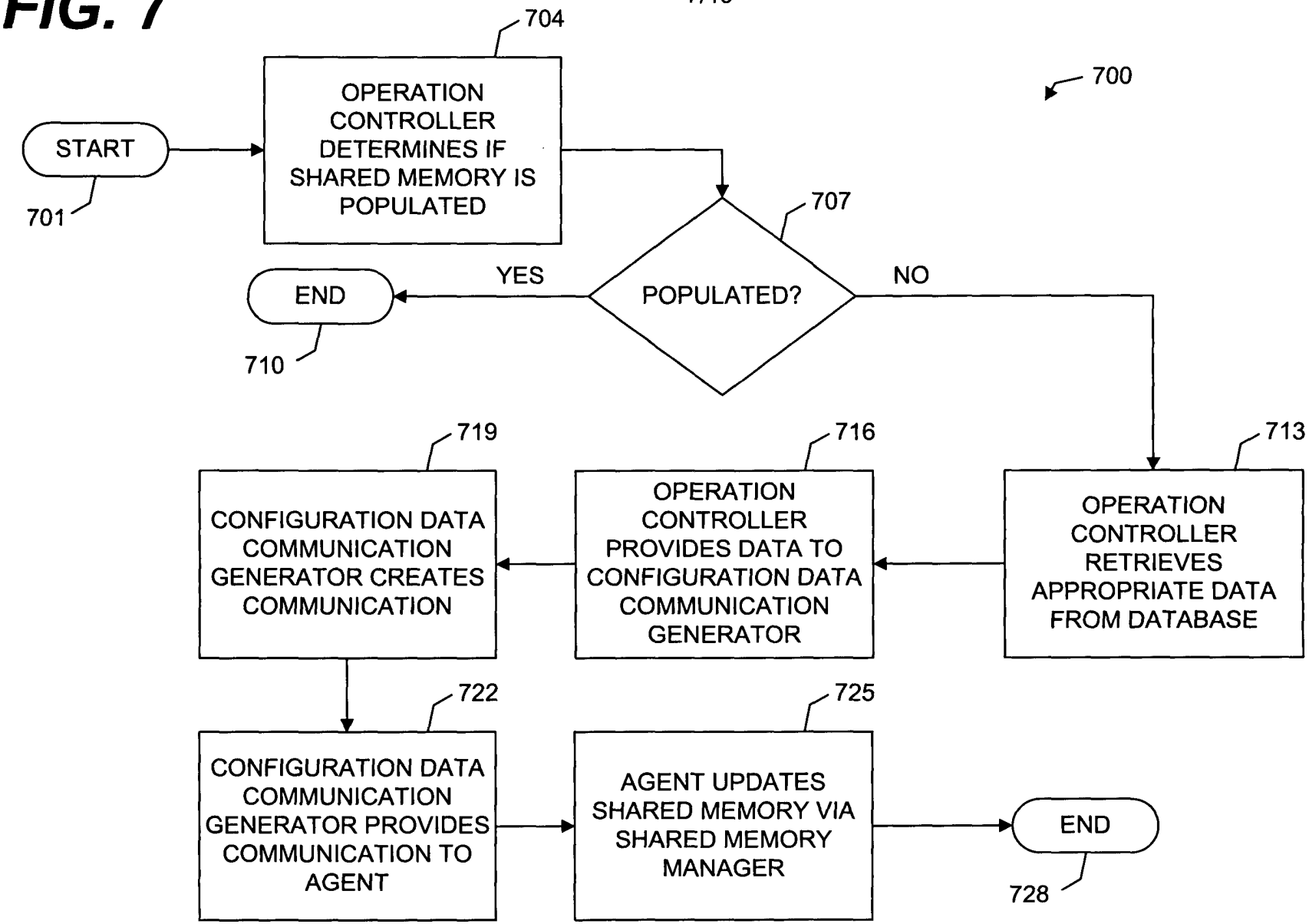

CONTROL MEMORY
<MEMORY TABLE A>
<MEMORY TABLE B>
<MEMORY TABLE Z>
524
527
MEMORY TABLE "A"
601a
<KEYS, MAXKEYS, VALUES, MAXVALUES, STRINGPOOL>
604a
OFFSET TABLE
607a
KEY TABLE
610a
VALUE TABLE
613a
STRING POOL
616a
MEMORY TABLE "Z"
601z
<KEYS, MAXKEYS, VALUES, MAXVALUES, STRINGPOOL>
604z
OFFSET TABLE
607z
KEY TABLE
610z
VALUE TABLE
613z
STRING POOL
616z

FIG. 6

FIG. 7


700
START
701
OPERATION CONTROLLER DETERMINES IF SHARED MEMORY IS POPULATED
704
POPULATED?
707
YES
NO
END
710
OPERATION CONTROLLER RETRIEVES APPROPRIATE DATA FROM DATABASE
713
OPERATION CONTROLLER PROVIDES DATA TO CONFIGURATION DATA COMMUNICATION GENERATOR
716
CONFIGURATION DATA COMMUNICATION GENERATOR CREATES COMMUNICATION
719
CONFIGURATION DATA COMMUNICATION GENERATOR PROVIDES COMMUNICATION TO AGENT
722
AGENT UPDATES SHARED MEMORY VIA SHARED MEMORY MANAGER
725
END
728

<?XML version='1.0'?>
801
800
<Envelope>
804
<Body>
807
<Jukebox>
810
<Genre Name="GenreName">
813
<Record Type="string"> string </Record>
816
816b
<Record Type="int"> 12345 </Record>
819
819b
</Genre>
813b
<Port name="portname" protocol="http" server="no" external="no">
822
<Wire> http://www.abc.com </Wire>
825
825b
</Port>
822b
</Jukebox>
810b
</Body>
807b
804b
</Envelope>

FIG. 8

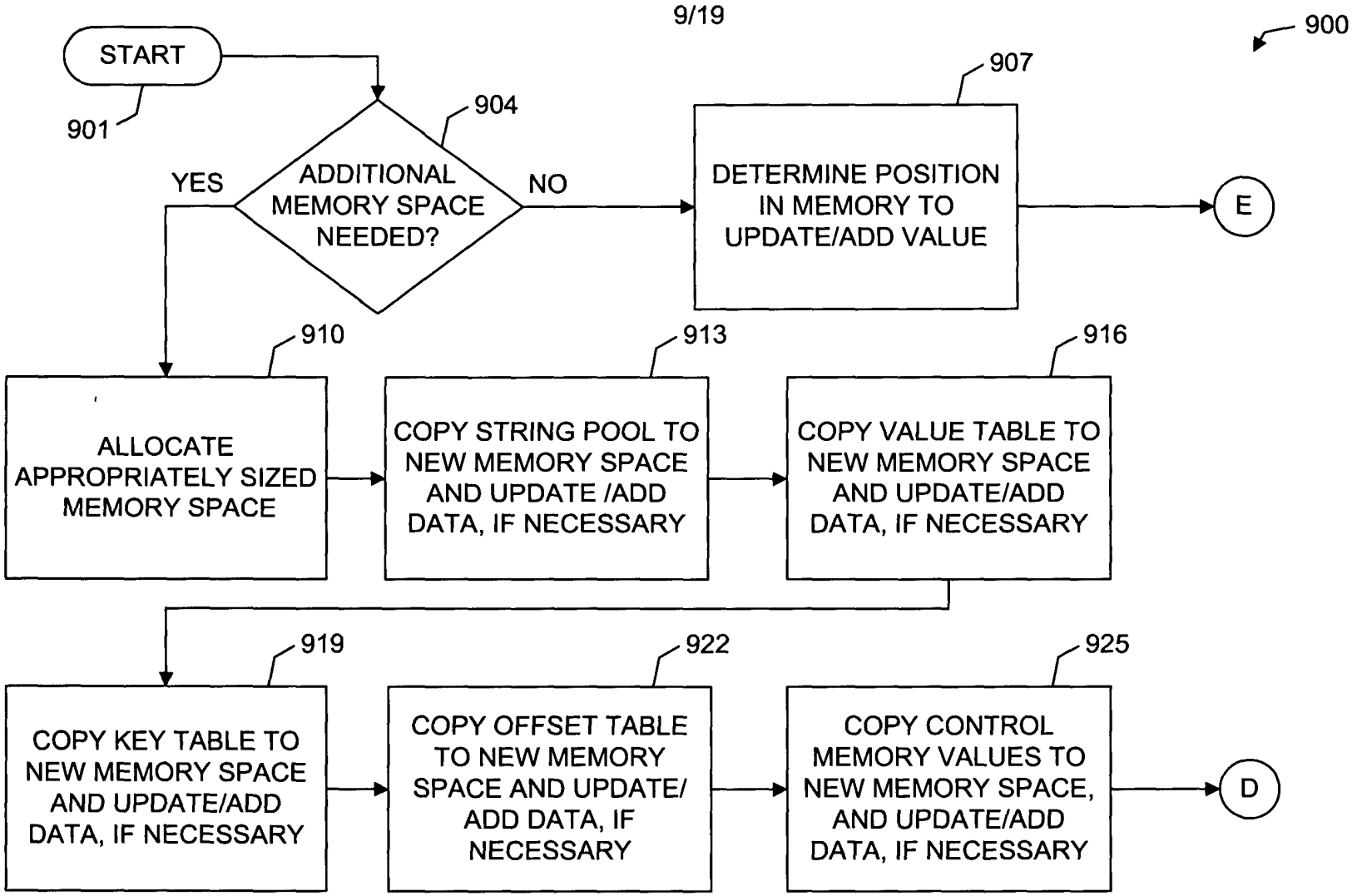

900
START
901
904
ADDITIONAL MEMORY SPACE NEEDED?
YES
NO
907
DETERMINE POSITION IN MEMORY TO UPDATE/ADD VALUE
E
910
ALLOCATE APPROPRIATELY SIZED MEMORY SPACE
913
COPY STRING POOL TO NEW MEMORY SPACE AND UPDATE /ADD DATA, IF NECESSARY
916
COPY VALUE TABLE TO NEW MEMORY SPACE AND UPDATE/ADD DATA, IF NECESSARY
919
COPY KEY TABLE TO NEW MEMORY SPACE AND UPDATE/ADD DATA, IF NECESSARY
922
COPY OFFSET TABLE TO NEW MEMORY SPACE AND UPDATE/ ADD DATA, IF NECESSARY
925
COPY CONTROL MEMORY VALUES TO NEW MEMORY SPACE, AND UPDATE/ADD DATA, IF NECESSARY
D

FIG. 9A

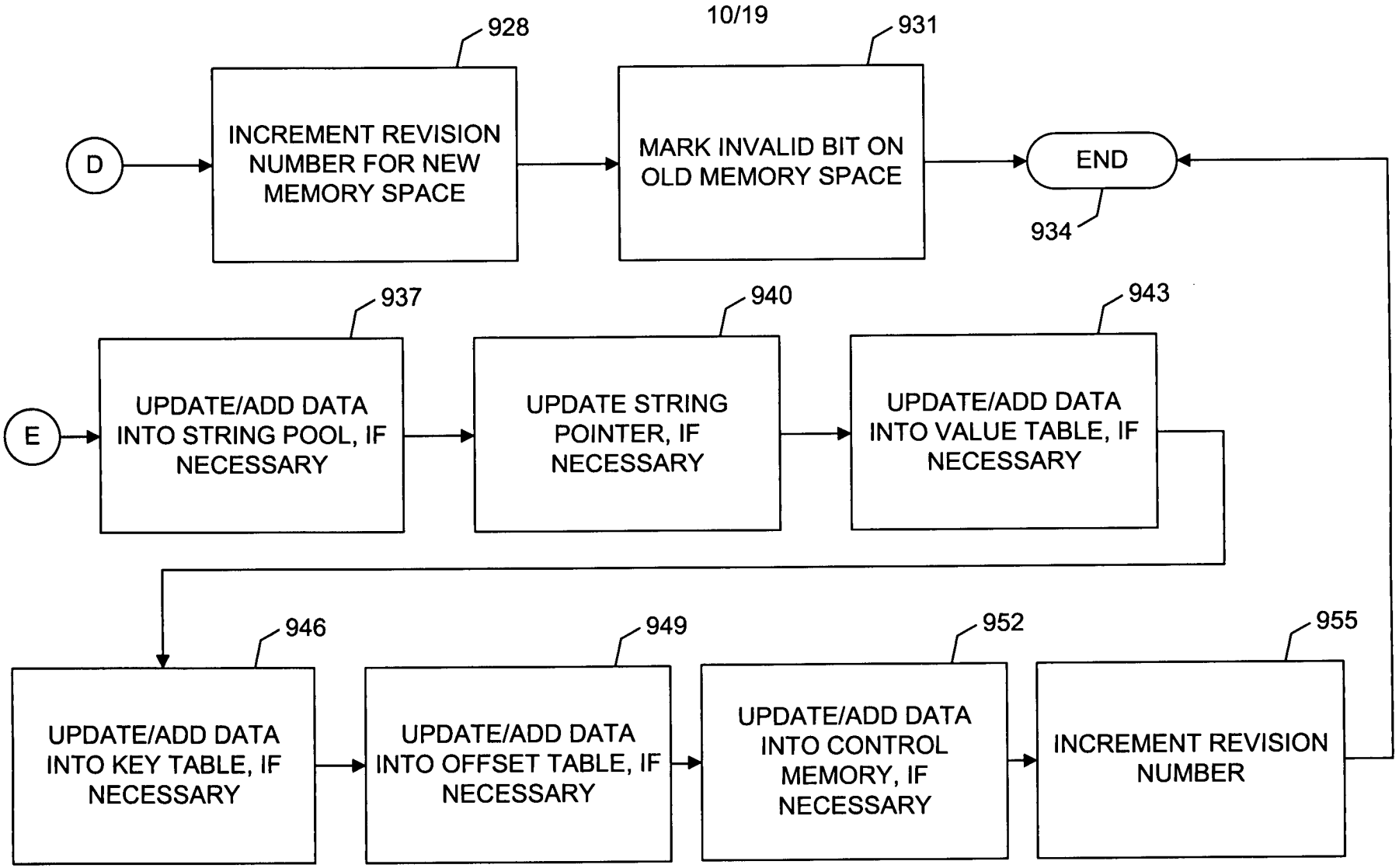

928
INCREMENT REVISION NUMBER FOR NEW MEMORY SPACE
931
MARK INVALID BIT ON OLD MEMORY SPACE
D
END
934
937
UPDATE/ADD DATA INTO STRING POOL, IF NECESSARY
940
UPDATE STRING POINTER, IF NECESSARY
943
UPDATE/ADD DATA INTO VALUE TABLE, IF NECESSARY
E
946
UPDATE/ADD DATA INTO KEY TABLE, IF NECESSARY
949
UPDATE/ADD DATA INTO OFFSET TABLE, IF NECESSARY
952
UPDATE/ADD DATA INTO CONTROL MEMORY, IF NECESSARY
955
INCREMENT REVISION NUMBER

FIG. 9B

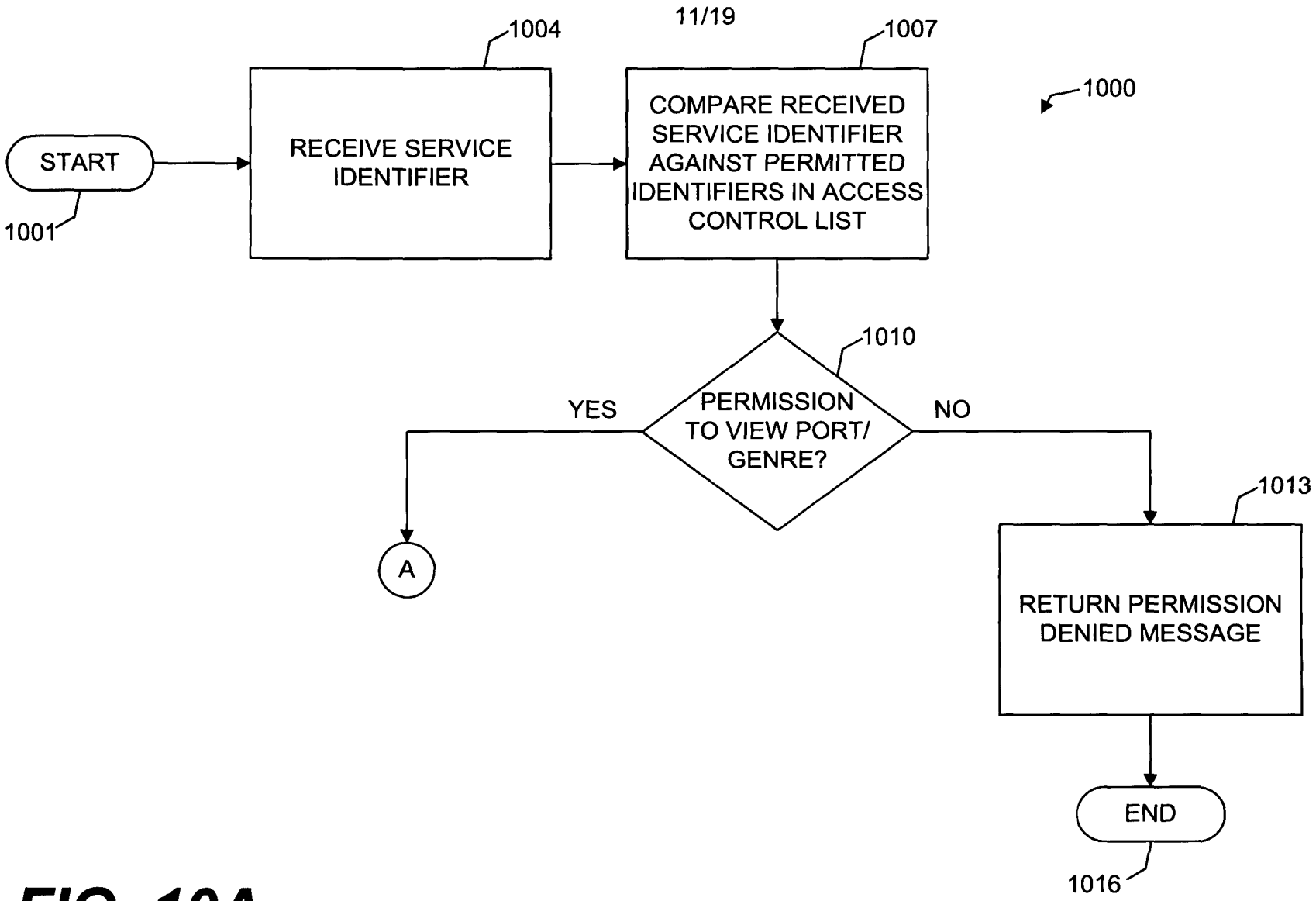

1004
1007
1000
START
RECEIVE SERVICE IDENTIFIER
COMPARE RECEIVED SERVICE IDENTIFIER AGAINST PERMITTED IDENTIFIERS IN ACCESS CONTROL LIST
1001
1010
PERMISSION TO VIEW PORT/ GENRE?
YES
NO
A
1013
RETURN PERMISSION DENIED MESSAGE
END
1016

FIG. 10A

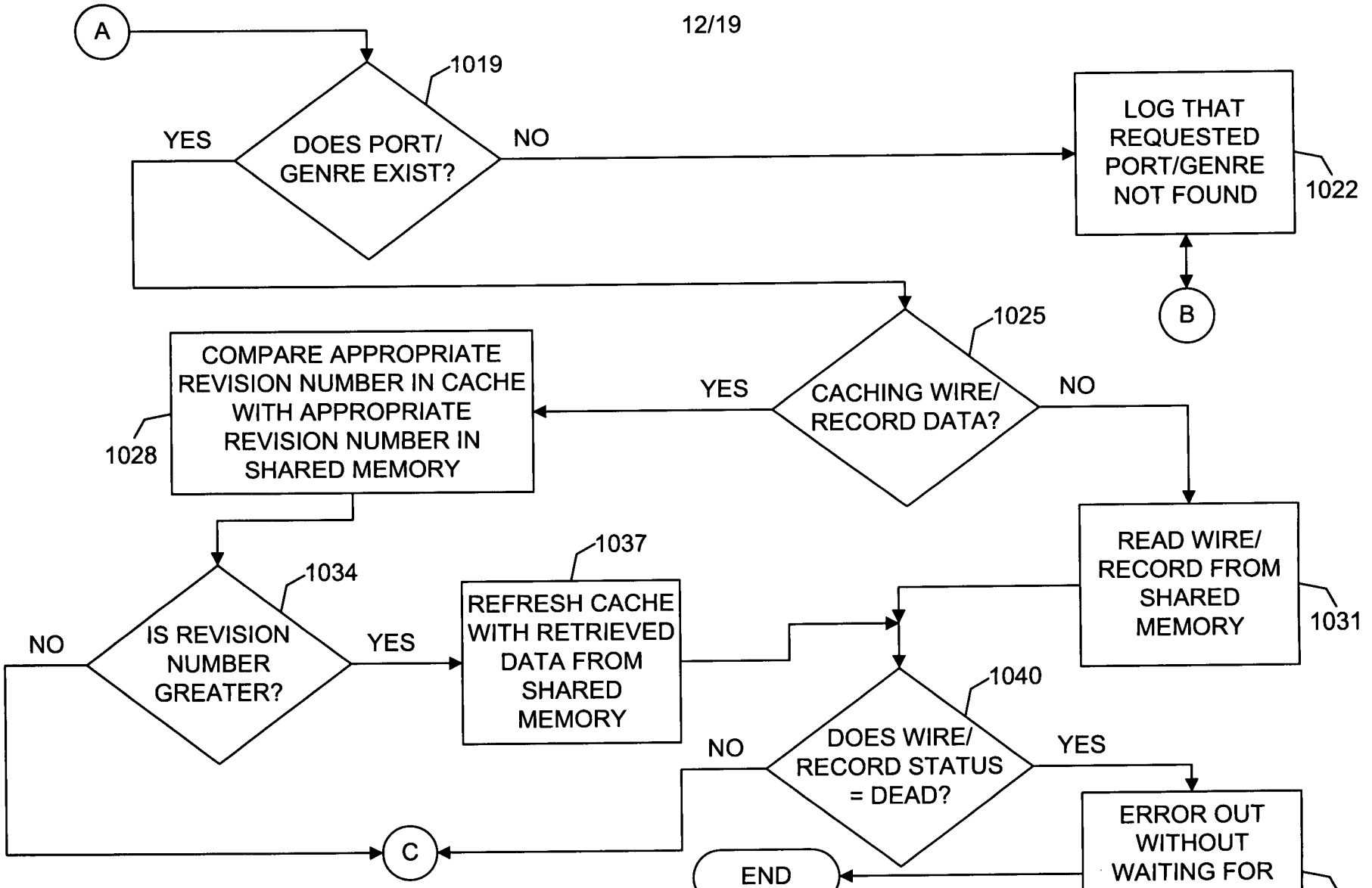

A
1019
DOES PORT/ GENRE EXIST?
YES
NO
LOG THAT REQUESTED PORT/GENRE NOT FOUND
1022
B
1025
CACHING WIRE/ RECORD DATA?
YES
NO
COMPARE APPROPRIATE REVISION NUMBER IN CACHE WITH APPROPRIATE REVISION NUMBER IN SHARED MEMORY
1028
READ WIRE/ RECORD FROM SHARED MEMORY
1031
1034
IS REVISION NUMBER GREATER?
NO
YES
1037
REFRESH CACHE WITH RETRIEVED DATA FROM SHARED MEMORY
1040
DOES WIRE/ RECORD STATUS = DEAD?
NO
YES
ERROR OUT WITHOUT WAITING FOR TIME-OUT FROM RESOURCE
1043
C
END
1046

FIG. 10B

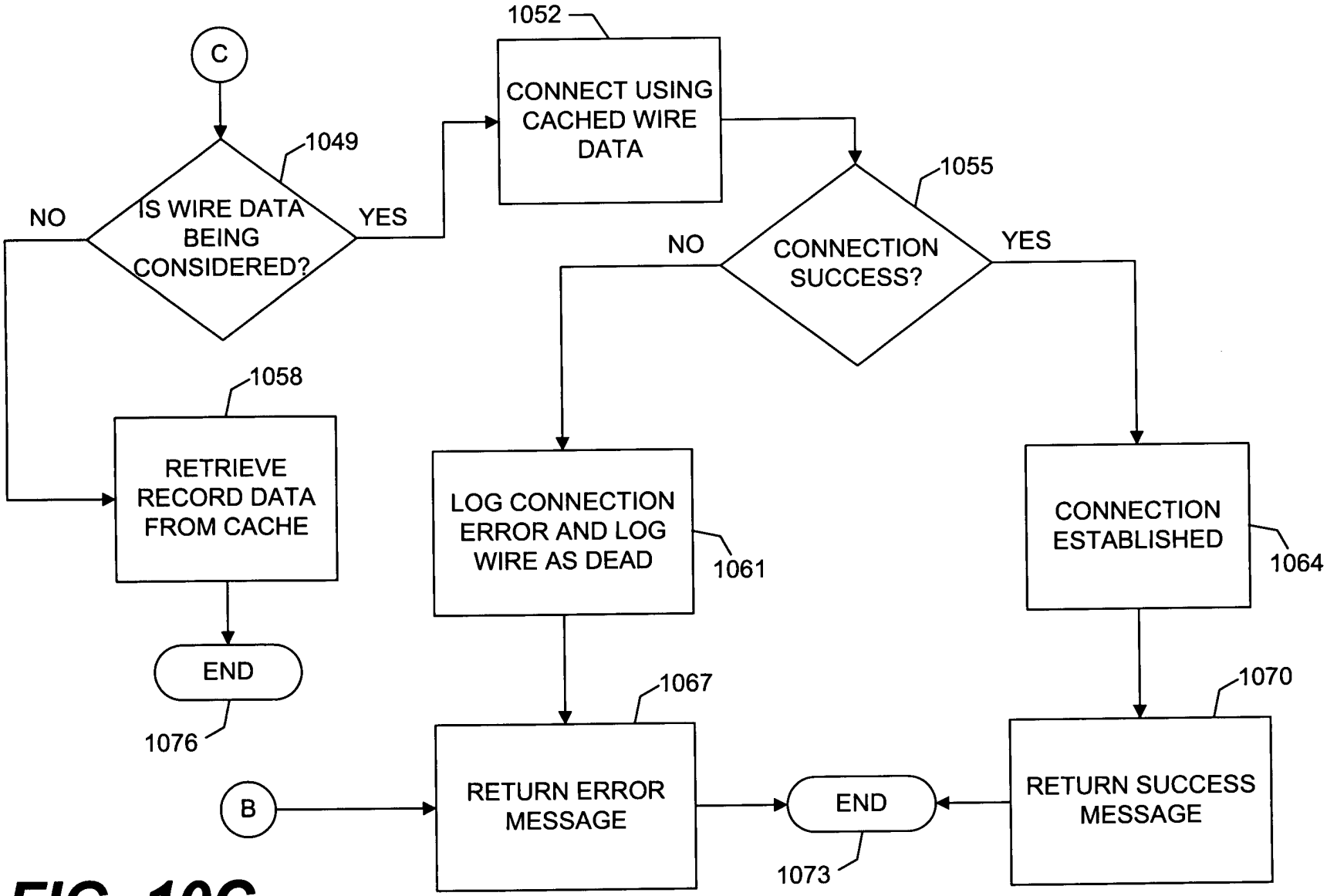

C
1049
IS WIRE DATA BEING CONSIDERED?
NO
YES
1052
CONNECT USING CACHED WIRE DATA
1055
CONNECTION SUCCESS?
NO
YES
1058
RETRIEVE RECORD DATA FROM CACHE
END
1076
LOG CONNECTION ERROR AND LOG WIRE AS DEAD
1061
CONNECTION ESTABLISHED
1064
1067
1070
B
RETURN ERROR MESSAGE
END
1073
RETURN SUCCESS MESSAGE

FIG. 10C

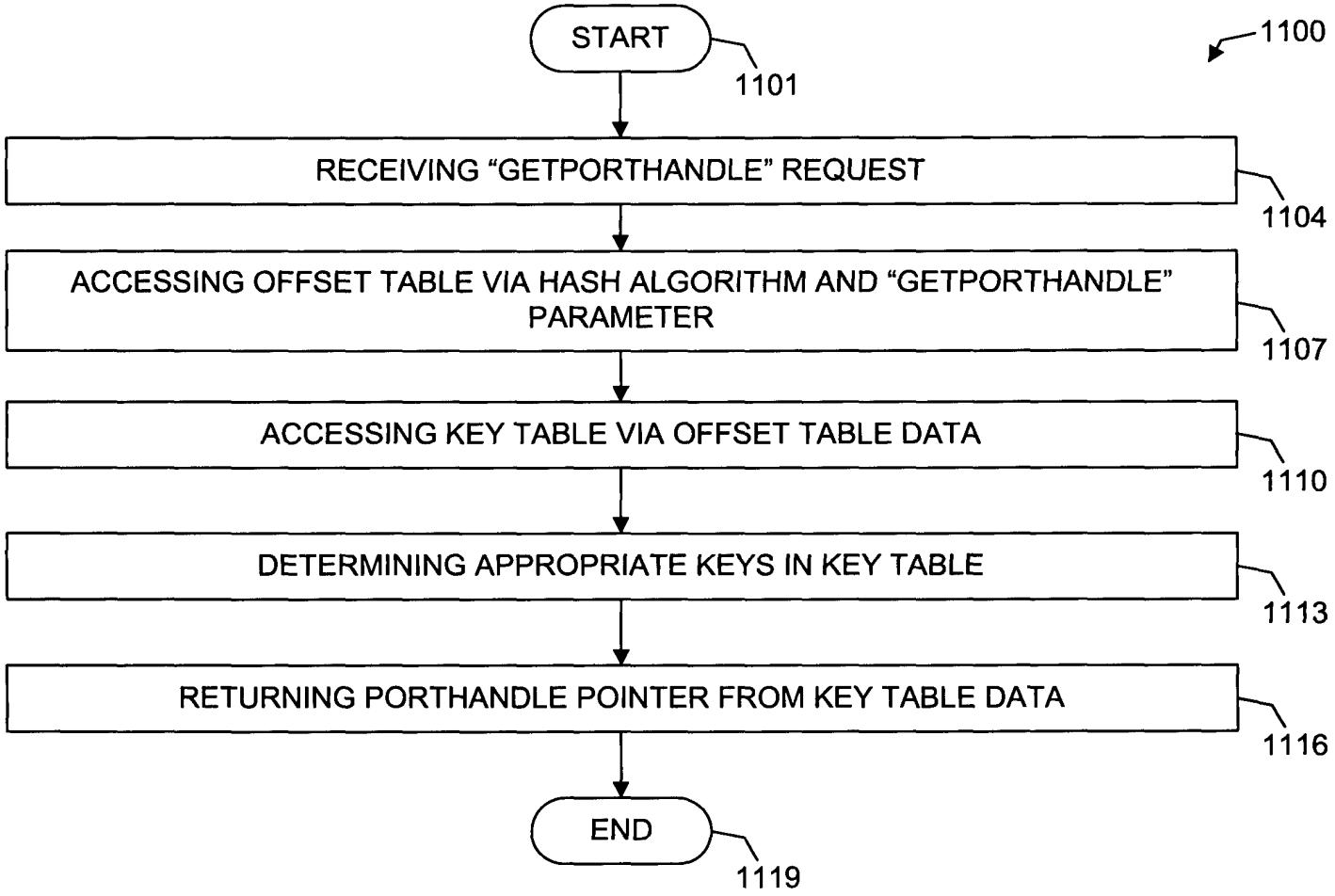

START
1101
1100
RECEIVING "GETPORTHANDLE" REQUEST
1104
ACCESSING OFFSET TABLE VIA HASH ALGORITHM AND "GETPORTHANDLE" PARAMETER
1107
ACCESSING KEY TABLE VIA OFFSET TABLE DATA
1110
DETERMINING APPROPRIATE KEYS IN KEY TABLE
1113
RETURNING PORTHANDLE POINTER FROM KEY TABLE DATA
1116
END
1119

FIG. 11

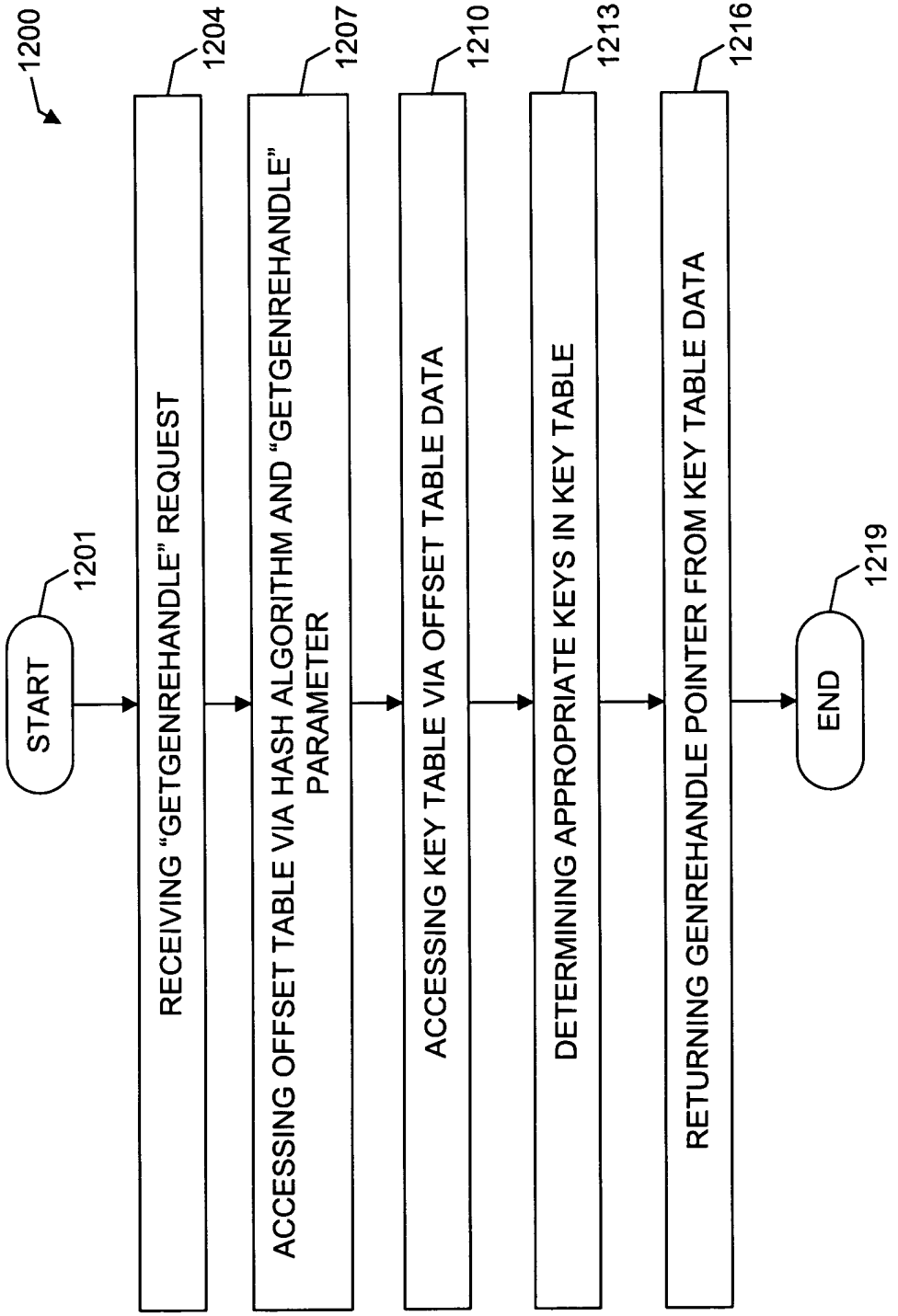

1200
START
1201
RECEIVING "GETGENREHANDLE" REQUEST
1204
ACCESSING OFFSET TABLE VIA HASH ALGORITHM AND "GETGENREHANDLE" PARAMETER
1207
ACCESSING KEY TABLE VIA OFFSET TABLE DATA
1210
DETERMINING APPROPRIATE KEYS IN KEY TABLE
1213
RETURNING GENREHANDLE POINTER FROM KEY TABLE DATA
1216
END
1219

FIG. 12

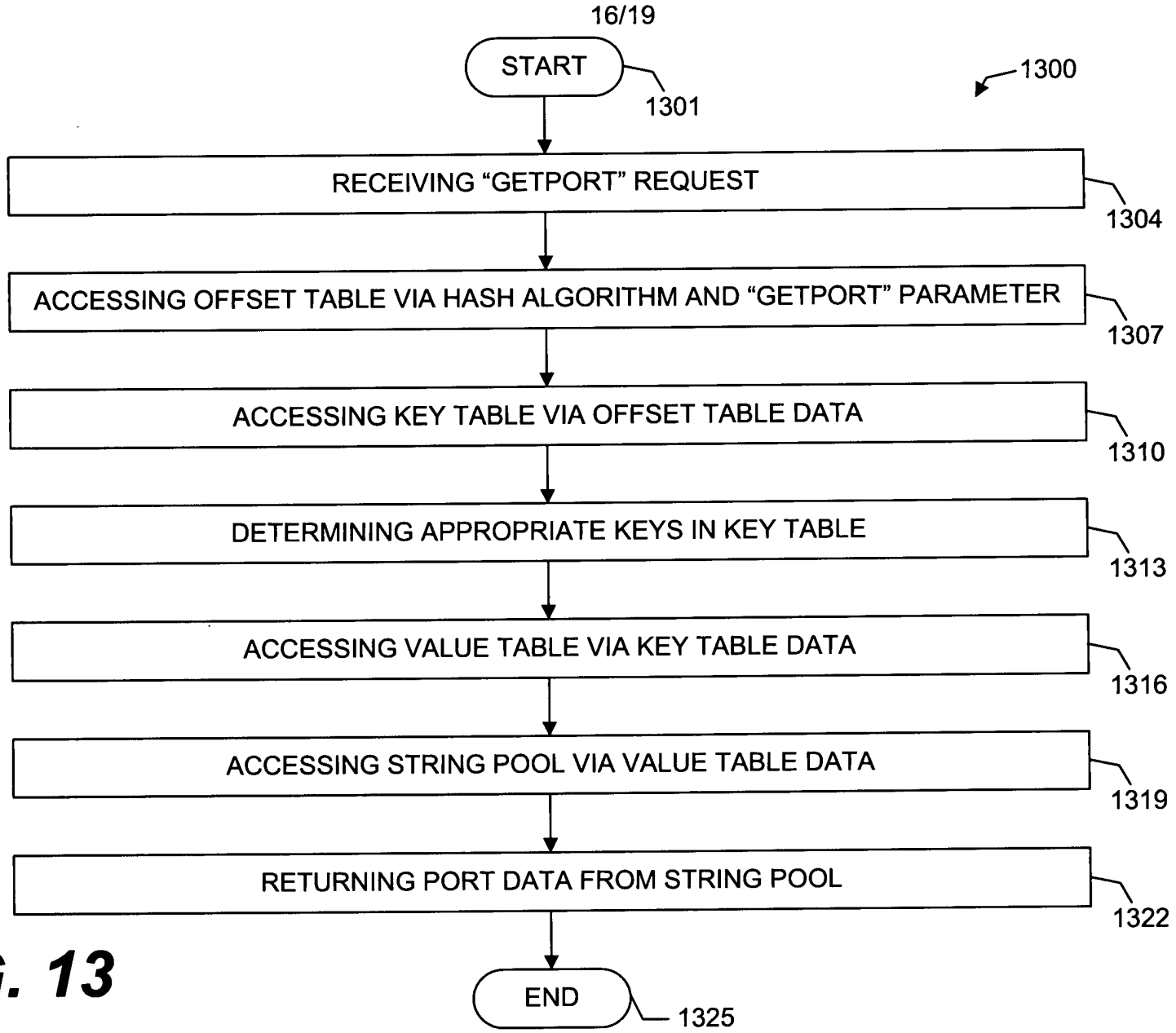

START
1301
1300
RECEIVING "GETPORT" REQUEST
1304
ACCESSING OFFSET TABLE VIA HASH ALGORITHM AND "GETPORT" PARAMETER
1307
ACCESSING KEY TABLE VIA OFFSET TABLE DATA
1310
DETERMINING APPROPRIATE KEYS IN KEY TABLE
1313
ACCESSING VALUE TABLE VIA KEY TABLE DATA
1316
ACCESSING STRING POOL VIA VALUE TABLE DATA
1319
RETURNING PORT DATA FROM STRING POOL
1322
END
1325

FIG. 13

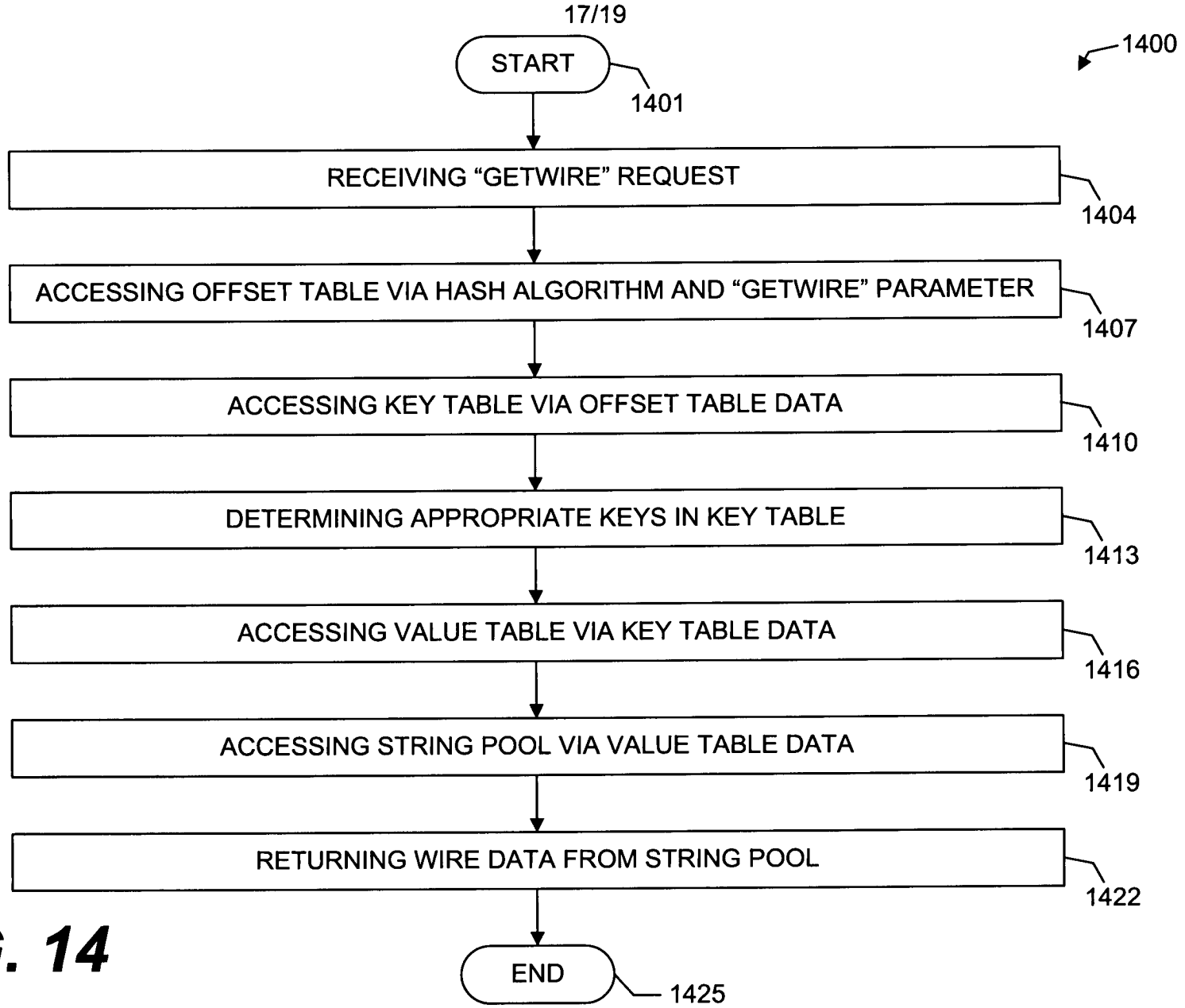

1400
START
1401
RECEIVING "GETWIRE" REQUEST
1404
ACCESSING OFFSET TABLE VIA HASH ALGORITHM AND "GETWIRE" PARAMETER
1407
ACCESSING KEY TABLE VIA OFFSET TABLE DATA
1410
DETERMINING APPROPRIATE KEYS IN KEY TABLE
1413
ACCESSING VALUE TABLE VIA KEY TABLE DATA
1416
ACCESSING STRING POOL VIA VALUE TABLE DATA
1419
RETURNING WIRE DATA FROM STRING POOL
1422
END
1425

FIG. 14

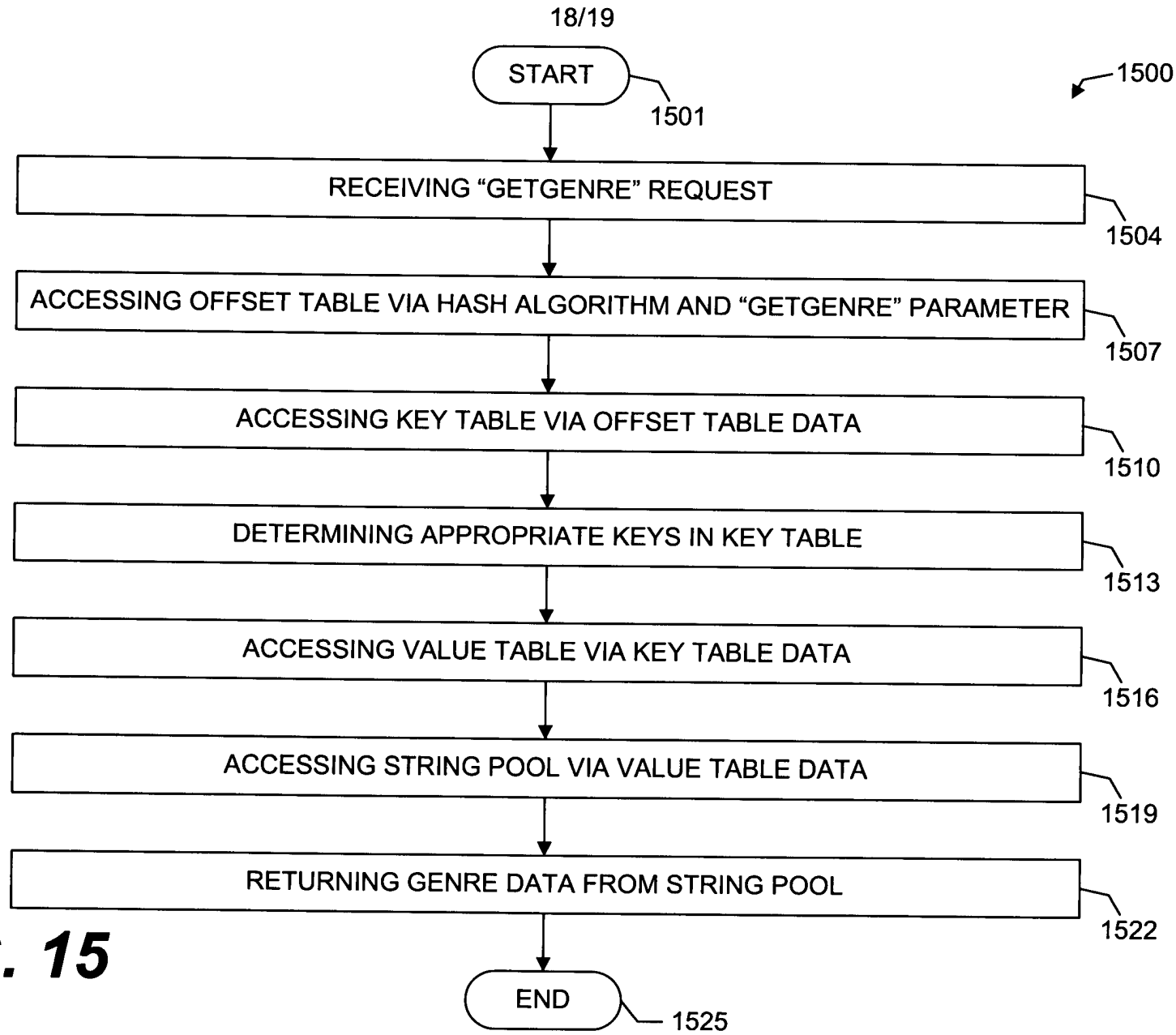

START
1501
1500
RECEIVING "GETGENRE" REQUEST
1504
ACCESSING OFFSET TABLE VIA HASH ALGORITHM AND "GETGENRE" PARAMETER
1507
ACCESSING KEY TABLE VIA OFFSET TABLE DATA
1510
DETERMINING APPROPRIATE KEYS IN KEY TABLE
1513
ACCESSING VALUE TABLE VIA KEY TABLE DATA
1516
ACCESSING STRING POOL VIA VALUE TABLE DATA
1519
RETURNING GENRE DATA FROM STRING POOL
1522
END
1525

FIG. 15

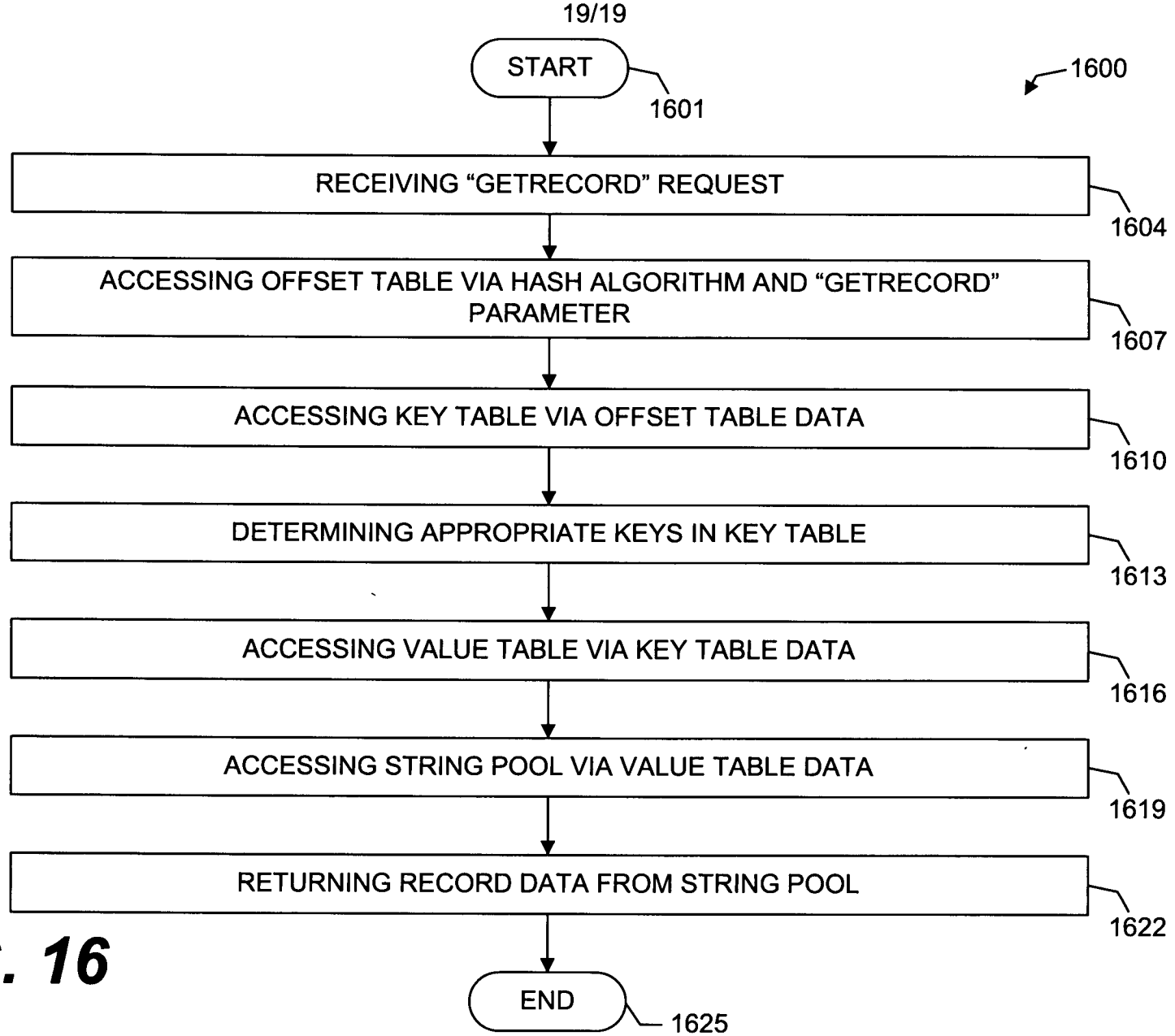

START
1601
1600
RECEIVING "GETRECORD" REQUEST
1604
ACCESSING OFFSET TABLE VIA HASH ALGORITHM AND "GETRECORD" PARAMETER
1607
ACCESSING KEY TABLE VIA OFFSET TABLE DATA
1610
DETERMINING APPROPRIATE KEYS IN KEY TABLE
1613
ACCESSING VALUE TABLE VIA KEY TABLE DATA
1616
ACCESSING STRING POOL VIA VALUE TABLE DATA
1619
RETURNING RECORD DATA FROM STRING POOL
1622
END
1625

FIG. 16